

Chapter 1

Junos Is Everywhere You Need to Be

In This Chapter

- Understanding the functions of a network operating system
 - Discovering how Junos OS is different
 - Looking beyond the operating system to the Junos Platform
-

The Junos operating system (Junos OS) is the software that runs networking and security devices from Juniper Networks. Administrators use Junos OS to set up devices and connect them together in a network, and dictate how the devices move, service, and secure traffic across the network. They also use it to monitor and, when necessary, restore the network.

Note: In this book, we use both Junos OS and Junos as one and the same.

This chapter introduces Junos OS by describing the functions of a network operating system and then discussing how they work in Junos. The chapter also explores key differences in how Juniper develops Junos software versus how other vendors create their network operating systems, and introduces additional components of the Junos portfolio.

Functions of a Network OS

Networks consist of specialized devices that pass along traffic from one to another. Each device must know what to do with each arriving bundle of traffic, or *packet*, so that the packet can continue its journey to its destination. The devices perform three primary functions to process each packet:

- ✓ Controls where the device sends the packets.
- ✓ Applies services such as prioritization or security.
- ✓ Forwards the packet to the next connecting device.

These actions are the primary functions of the network operating system that runs on the device. In simplest terms, the *control plane* of the network operating system is the brain of the device with the *forwarding plane* providing the brawn to quickly move packets through the system. Depending on the type of packet, the *services plane* may also provide packet services such as address translation, prioritization, and security.

Control functions

This essential map for connectivity, security, and other orchestrating processes is the function of the network operating system's control plane. The processes and information of the control plane must provide answers to two essential questions:

- ✔ How does the network direct the delivery of packets from one place to another? In other words, what are the routes or paths to establish, how do they change, and how does each device know which route to use for each packet?
- ✔ What does the network do with each of the packets along its journey? In other words, what are the handling rules, or policies, along with the security and services plane established for traffic delivery?

Although the questions can be simply stated, the possible responses are virtually limitless. You can define dozens and dozens of protocols to answer these questions for different types of network maps and all the different types of traffic, not to mention how the control plane monitors and manages everything. The many processes to control the network delivery of packets fill the industry with all those three- and four-letter acronyms that you've somehow managed to file into your memory.

Service functions

The service needs of users grow with new applications triggering new requirements for quality, security, addressing, and content delivery, among others. Applying the specialized processing is the role of the services plane.

For example, as the packets flow through each device, the devices must typically apply a range of filters, policies, and services for protecting the network (and its clients and applications) and assigning priorities for the use of its resources. Visualize watching a YouTube video. Now visualize all your users watching the video all at the same time because some clown in your office passed along an e-mail with the link to everyone. And now think about all the traffic hitting your network all at once, just as your president is on a critical call with your biggest customer. Oofta!

This is just one example of where you may want to define a few of those extra rules that your network can follow in making its packet deliveries. (We provide Chapter 15 to help you set up class of service in your devices.)

Forwarding functions

Along with assembling the intelligence to properly deliver the traffic from one place to another and applying services, the network operating system (and its hardware) must actually deliver packets to the correct destination using this intelligence. Moving packets through a networking device is the function of the network operating system's forwarding plane, also sometimes known as the *data plane*.

Packet forwarding takes care of the handling required to move each packet quickly from its inbound device interface to the proper outbound interface(s). For large networking devices that carry terabits of traffic, this handling must occur at an ultra-über fast rate to maintain the high packet throughput of the machines.

Taking Advantage of One Network OS

Network operating systems have a lot to do and can have a big impact on the performance, ease of operations, reliability, and security of your network. Junos OS is different, in that it's one operating system. But why does having one operating system matter?

One operating system means the Juniper engineers build upon the same set of code and then share this code, as appropriate, across all the platforms running Junos. For example, enterprise platforms use the same hardened implementation of the routing protocol Open Shortest Path First (OSPF) that has been running in large service provider networks for many years. It's not a different code set, but the same one. (To set up OSPF, see Chapter 10.)

So, if your responsibilities include administrating the network, you find that many features are configured and managed in the same way on the different platforms, whether they are routers, switches, or security devices. One operating system, therefore, saves you time, potentially lots of it, in everything from training to setup to ongoing operations. Also, if you plan changes in the network, one operating system can save you time there, too. With far less variation to evaluate, test, and deploy, it's less effort for feature roll out, software upgrades, and other network modifications.

Taking a Peek Inside the Junos OS

How engineers design a network operating system impacts the reliability, security, scalability and performance of not just the devices, but also the overall network, particularly in large-scale systems. The operating system must handle the many different processes essential to running today's global networks, while also assuring fair sharing of resources so that no process or service can starve out others.

World-class architecture

The architecture of the Junos operating system cleanly divides the functions of control, services, and forwarding into different planes. The control and services planes include many different processes that run in different modules of the operating system. The explicit division of responsibility allows the software to run on different engines of processing, memory, and other resources. This division of labor is what enables Junos to run all types of platforms in all matter of sizes, from a small box in a home office to the largest boxes in the world handling terabits of data every second.

Figure 1-1 provides a high-level view of the Junos OS software architecture with its three functional processing planes. Shown above the dashed line is the control plane that runs on what is known as the *Routing Engine* (RE) of the Juniper device. Below the dashed line is the packet forwarding plane, which runs on a separate *Packet Forwarding Engine* (PFE) in larger Juniper platforms. The services plane, which provides specialized processing, such as for quality classification and security, is on the right.

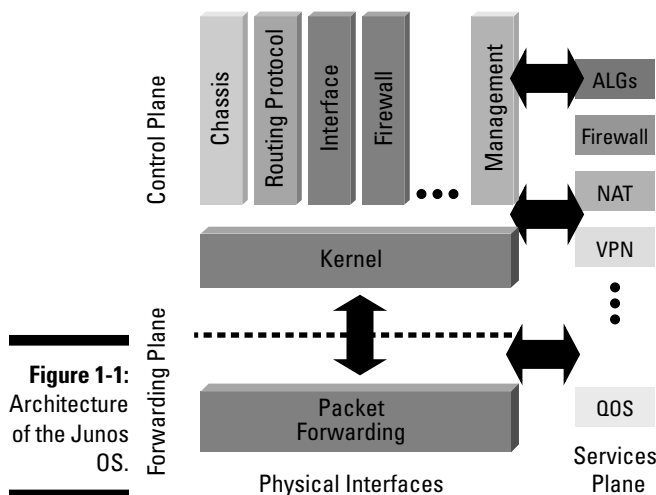


Figure 1-1:
Architecture
of the Junos
OS.

Do you want faster platforms in your network? That's like asking if you'd like to have today off (with pay, of course). Yes, it would be good to have the network go a little faster. Faster, faster, faster is a constant drumbeat for networks. In over ten years of product delivery, Juniper has scaled the throughput of its fastest devices from 40GB per second to multiple terabits per second with a fast expanding set of services. The use of separate processors for the RE, the PFE, and services cards has been the essential architecture element to each performance breakthrough. In particular, separation lets the PFE and services throughput follow in lock-step with the increasing speeds of the custom Application-Specific Integrated Circuits (ASICs) on which the PFE and services run in the largest platforms.

Separating the engines also reduces interdependencies between them. Not only does this separation help preserve the operation of each when another is experiencing problems, it also gives the Juniper engineers more ways to provide system redundancy and failover. For example, you find dual REs in some platforms, whereas the EX Series Ethernet Switches offer a capability called Virtual Chassis to provide redundancy, among other benefits. (See Chapter 11 for the details of this switching feature.)

Plain smart: The planes of Junos OS

Each of the planes of Junos OS provides a critical set of functionality in the operation of the network.

It's all under control

All the functions of the control plane run on the Routing Engine (whether you have a router, switch, or security platform running Junos). Figure 1-1, shown earlier in this chapter, shows the high-level design of the control plane — a set of modules, with clean interfaces between them, and an underlying kernel that controls the modules and manages all the needed communication back and forth among all the components. The kernel also handles the RE communications with the Packet Forwarding Engine and the services. Each of the different modules provides a different control process, such as control for the chassis components, Ethernet switching, routing protocols, interfaces, management, and so on.



The basis of the Junos kernel comes from the FreeBSD UNIX operating system, an open source software system. This mature, general-purpose system provides many of the essential functions of an operating system, such as the scheduling of resources. To transform it into a network operating system, the Juniper engineers extensively modified and hardened the code for the specialized requirements of networking.

You may be wondering if you have a way in Junos OS to protect the control plane itself from a security attack. Yes, you can configure filters and rate-limit the traffic that reaches your RE. (For more on this topic, see Chapter 9.)

Moving forward

The Packet Forwarding Engine is the central processing element of the forwarding plane, systematically moving the packets in and out of the device. In the Junos OS, the PFE has a locally stored *forwarding table*. The forwarding table is a synchronized copy of all the information from the RE that the forwarding plane needs to handle each packet, including outgoing interfaces, addresses, and so on. Storing a local copy of this information allows the PFE to get its job done without going to the control plane every time that it needs to process a packet.

Another benefit to having a local copy is that the PFE can continue forwarding packets, even when a disruption occurs to the control plane, such as when a routing or other process issue happens.

At your service

The services plane provides special handling required by many different types of packets. By separating the processing of services from other functions of the operating systems, Junos OS is able to support a wide variety of different service types in different kinds of platforms.

These services might include prioritizing a packet carrying time-sensitive information, such as a voice call, ahead of others on a congested link; guarding which users can get to what sections or applications of the network; translating addresses where one network meets another; or mediating how the network serves video content.

That's not a problem: The many benefits of modular architecture

Have you ever had a router continually reboot, and when you look on the console, you see that an error occurred in a single nonessential process?

With the Junos OS, you don't see that problem. The modular architecture of Junos OS allows individual control plane processes to run in their own *module* (also sometimes called a *daemon*). Each module has specified processing resources and runs in its own protected memory space, avoiding the processing conflicts that can occur in other platforms. If a malfunction in a module causes an issue, the rest of the system can continue to operate. For example, one module can't disrupt another by scribbling on its memory.

What about a minor hiccup in SNMP bringing down your whole system? That's another misfortune that you won't miss with Junos OS, because its clean separation between control processes helps to isolate small problems so that they can't create worse havoc.

In our many discussions with users, we hear over and over that the stability of Junos OS is the biggest difference that they see after deploying Juniper platforms in their network. They tell us about their boxes running for months and months, even years without interruption. How they popped the device into the rack, set up the configuration, and never looked back. It just keeps going and going and going — oh, that's another company's line.

The modular architecture also eases fault isolation. With each process functioning within its own module, when the occasional problem does occur, pinpointing the exact reason is far less complicated for both you and the Juniper support team. With quick identification and a good understanding of the root cause, you can apply a fix that works, the first time you try.

We have one more benefit to highlight — flexible innovation. The organized structure of the architecture enables deep integration of new capabilities with high functioning interaction with existing processes. For you, this means that native support of new services and features delivers a richness of capability with the high performance that you expect. For example, among the integrations to Junos OS are the security services derived from Juniper's ScreenOS operating system. (You can find out more about some of these features in Chapter 12.)

Developing Junos OS

Software development probably isn't a topic that you expected to find in a networking book. After all, you don't need to build the Junos operating system. Juniper's engineers do that for you.

However, we include a little about this topic because we think it's important. The disciplined development process is an essential reason why Junos OS is different. Unlike most other vendors, Juniper develops new versions of the operating system along a common release path, as shown in Figure 1-2.

Figure 1-2:
Junos
release
path.



Asking the right questions

Consider reviewing the software development processes of vendors as a part of your evaluation of new network and security equipment, because it can save you time and money down the road. Here are some questions that you can ask vendors about their software development processes:

- ✓ **Software versions:** Find out how many different software versions exist for the products you are buying, and ask why the different versions exist. Know the differences between versions, and when to use one version versus another. Also, ask about the support and end of life policies for each version.
- ✓ **New features:** Ask what steps do development engineers follow when adding new features. How do they support changes and fixes to the features in different software versions or release trains? You also want to know how they decide which features to add to which version?
- ✓ **Fixes:** Ask about the steps for adding fixes to the code. What procedures ensure that a new fix is a part of all releases, including those in the future? In what types of releases are fixes available and how often?
- ✓ **Testing:** Find out how the vendor tests newly developed features (and fixes). What guidelines determine when a release is ready for customers? How thoroughly is each type of release tested before being released to customers? Ask whether a new release can affect previously working features, and find out whether the vendor's testing includes use-cases to assess how features interact. Ask how the vendor performs performance and scalability testing.

When it's time to upgrade, you simply choose a higher release number, and not only do you get all the newly developed features, you also keep all the important features you've already been using in your network. Also, you're not running a specialized version of software that may be prone to issues; you're running software used by Juniper customers everywhere. (Find out more about Junos release versions and upgrades in Chapter 3.)

Beyond the OS

Since we wrote the first edition of this book, Juniper has expanded its Junos software portfolio beyond the operating system, adding new capabilities to link into the application space as well as client software for mobile and personal computing devices. Together, the Junos operating system, the Junos Space network application platform, and the Junos Pulse client form the Junos Platform. By integrating these software layers of the network into one platform, Juniper is expanding the ways that applications can interact with the network from the cloud out to the end user.

As part of the Junos Platform, Juniper provides a set of programming interfaces and software development kits (SDKs) that developers can use to specify the application interactions. Unlike other platforms that merely enable third parties to interface through APIs, these SDKs give application developers a broad set of development interfaces and tools to build a wide variety of applications richly integrated to the Junos Platform.

The following sections provide a short introduction to these additional components of the Junos portfolio. To learn more about Junos Space and Junos Pulse visit <http://www.juniper.net/us/en/products-services/software/junos-platform>.

Junos Space

Junos Space is an open network platform for developing and hosting applications that interact with the network. The Junos Space platform provides multilayered network abstractions and workflows that allow users to automate network operations and increase operator efficiency.

The software includes a scalable runtime environment with multitenant, hot-pluggable network application support, a network application development framework, and a Web 2.0 user interface.

Junos Space provides a development environment for fast development of network-aware applications. The application development framework includes a common infrastructure, a software development kit (SDK) with prebuilt core services and widgets to allow easy user-interface prototyping, and standards-based APIs for third-party application integration. Using the Junos Space SDK, developers have the option of creating different classes of applications. These include mashups, customized business process workflows, and native applications.

Junos Pulse

Junos Pulse is an integrated, multi-service network client to secure and control application delivery on mobile and personal computing devices. The client is both identity- and location-aware, enabling seamless migration from one access method to another so that users effortlessly, yet securely, retain access to their applications regardless of their location.

Users can download the Junos Pulse client free of charge from most mobile application stores. In this way, the solution can support not only devices

managed by the network provider, but also those managed by the user. Once installed, the client provides secure, authenticated access to corporate networks and resources using SSL VPN technology. Junos Pulse provides a comprehensive mobile security, management, and control solution to protect end devices from viruses, malware, spam, loss, theft, physical compromise, and other threats.